The *Mid*-Infrared
*I*nterferometric
Instrument for the
Very Large Telescope
Interferometer

MIDI

—

Software User and Maintenance Guide: Overview

User Manual

Doc. No. VLT-TRE-MID-15824-0262
Issue 1.5.235
Date August 23, 2007

Prepared Richard J. Mathar, Walter J. Jaffe August 23, 2007
Name Date Signature

Approved U. Graser
Name Date Signature

Released Ch. Leinert
Name Date Signature

Change Record

Issue	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
1.0	27-July-2002	all	first preparation
1.1	29-May-2003	6,3.2	added
			merged "Reference Docs." with "References"
			removed empty List of Tables
			corrected cross-references in Fig Captions
	30-May-2003	4.3.1	added
	11-Jun-2003	Fig. 1	shows 1-Computer DCS solution
		Sec. 4.2, Fig. 6	added
	26-Jun-2003	4.1.1	added hint on .netrc and .pecs installation
		4.3.5	added
	27-Jun-2003	4.1.4	added
	30-Jun-2003	4.1.5, 4.3.6	added
	15-Jul-2003	4.3.7	added
	17-Jul-2003	5.1.2	added
	05-Nov-2003	4.3.3	added
	19-Nov-2003	3.1.3	changed name of NRTS default server
	20-Nov-2003	5.1.1	added
1.2	27-Nov-2003	4.3.8, 4.3.4	added
	05-Dec-2003	6	update man page listings
	11-Dec-2003	4.1.1	updated
	23-Jan-2004	4.1.5	updated for P2PP 2.7
1.3	21-May-2004	1.1	update URL's
	09-Jul-2004	4.1.1	updated w.r.t vltisimBUILD
		3.2	removed section on vltisim chopping
	10-Sep-2004	4.3.8	updated
		4.3.9	added
	21-Sep-2004	4.1.5	updated for P2PP 2.8
1.4	30-Oct-2004	4.1.1	removed comments on `fv`
	25-Nov-2004	6	removed man-page printouts
1.5.235	23-Aug-2007	all	produced PDF version
		6	partially updated

Contents

List of Figures

1 INTRODUCTION

MIDI is a Michelson-type interferometric instrument to be operated at 10 μm at the VLTI on Paranal, Chile. The instrument's most challenging tasks from a control point of view are the high detector readout frequencies (40–250 Hz) which will be processed in a (near) real-time environment. The wish for setting up a feed-back system with an open-loop frequency of up to 10 Hz to stabilize fringes (when no external fringe tracker is available), sets increased demands on data acquisition, real-time processing, and near real-time processing capabilities. Accurate synchronization among many different sub-systems (chopping control systems, fringe tracker, delay line controllers, etc.) introduces further strong requirements.

1.1 Scope

This document is the introduction to the MIDI Software User and Maintenance Manuals. Separate manuals have been written for the major MIDI software subsystems: OS, ICS, DCS, NRTS and for the Observation Templates. The current document summarizes the software architecture of MIDI, describes the differences between MIDI and earlier "standard" VLT instruments, and describes the procedures and GUIs for starting up and shutting down the system as a whole. Detailed descriptions of the procedures and GUIs for the subsystems are described in their separate documents.

This report contains the following sections:

INTRODUCTION The present section, which states the aim of this report and lists the references as well as the used abbreviations and acronyms.

OVERALL ARCHITECTURE

SYSTEM USER MANUAL Commands and GUIs for starting up, running, and shutting down MIDI as a whole.

MAINTENANCE MANUAL Procedures for installing the MIDI software.

New versions of this document are distributed

- on the world wide web under `http://www.mpia.de/MIDI/internal/vlt_docs.html`
- in the CMM module `miins`, directory `miins/doc`, therefore also on the IWS in `~midimgr/$UNAMEN/miins/doc`,
- in the CMM module VLT-TRE-MID-15824-0262 ,
- on the world wide web under `http://www.strw.leidenuniv.nl/~mathar/public`.

1.2 Abbreviations and Acronyms

ACC	Access and Configuration Control
ATM	Asynchronous transfer mode
CE	Calculational engine
CMM	Configuration Management module
DCS	Detector control system
DL	delay line
DWS	Detector Workstation
ESO	European Southern Observatory `http://www.eso.org`
GUI(s)	Graphical User Interface(s)
GEIRS	Generic infrared software
HP	Hewlett Packard
ICS	Instrument Control Software
IRACE	Infrared array control electronics
ISF	Instrument summary file
IWS	Instrument Workstation
LAN(s)	Local Area Network(s)
LCU(s)	Local Control Unit(s)
MIDI	Mid-infrared Interferometric Instrument for the VLTI `http://www.mpia.de/MIDI`
MPIA	Max-Planck-Institut für Astronomie, Heidelberg `http://www.mpia.de`
NRTS	Near Real Time System
OPD(s)	optical path difference(s)
OS	Observation software
PLIN	point list index number
ROE	Readout electronics
SUN	Sun microsystems
UWS	User workstation
VLTCS	VLT Common Software `http://www.eso.org/vlt/sw-dev`
VLTI	Very Large Telescope Interferometer `http://www.eso.org/projects/vlti`
WS(s)	workstation(s)

2 MIDI SYSTEM ARCHITECTURE

2.1 Data Processing Requirements

MIDI is equipped with a mid-infrared sensitive detector which can — together with its controlling readout electronics (ROE)— deliver data at rates as high as about 53 MB/s (burst mode). Normal observing does not require these rates, but average rates of 2.3 MB/s are foreseen, along with a total data volume of 41 GB/night. These video data rates require powerful instrument hardware to collect and compress the data before archiving.

In order to efficiently acquire reliable interferometric data MIDI has to track the atmospheric contributions to the fringe motion at rates of $\sim$ 1 Hz, i.e. in "Near Real" time.

These requirements have led to "non-ESO-standard" elements of MIDI data processing design, described in more detail in documents [4, 5, 6, 7]. Additionally, for pragmatic reasons during system development, the DCS system does not use the ESO IRACE system but rather the Heidelberg GEIRS system of hardware and software to control the detector electronics. We summarize the non-standard elements here.

- The DCS system is subdivided into three subsystems:

1. The DCS Server, resident on the Instrument Workstation (IWS) which provides a central, ESO-style communications point with the other subsystems
2. GEIRS, which controls the ReadOut Electronics and collects data from the detector, and is largely resident on the Detector Workstation (DWS)
3. NRTS, the Near Real Time System, which receives "windowed" data from GEIRS via a high speed parallel connection, and prepares it for archiving and is resident on the Post Processing (PP) computer `wminrts`. It also performs online so-called "Data Quality Analysis" of which the most important is fringe tracking.

- The ICS system must control the fringe piezos in real time, via the MIDI Delay Line LCU. The exact timing and position of piezo motions and telescope chopping movements must be known to the NRTS both for fringe tracking and for inclusion in the archived data. This has as consequences:

  1. The Observing System (OS) downloads a preplanned *Schedule* of piezo movements to the DL-LCU and to NRTS during observation SETUP via the ESO CCS messaging system
  2. The DL-LCU forwards actual piezo positions, exposure timing and chopping mirror timing information to NRTS, again via CCS messaging.

Figure 1 illustrates the hardware components of the MIDI computer system. For more details, see [4].

Figure 2, also taken from [4] illustrates the flow of commands, image data, and OPD/timing information between software systems.

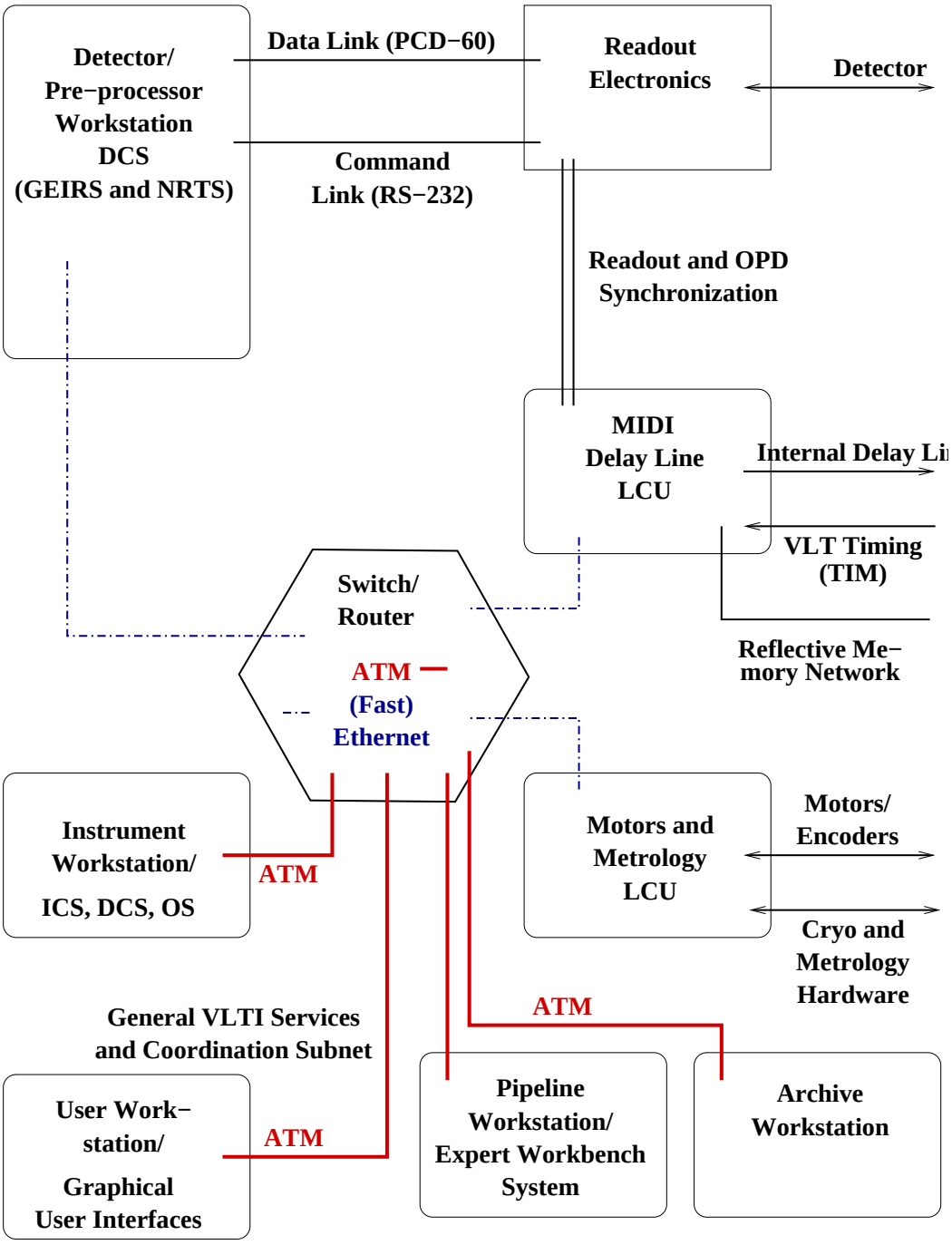

Figure 1: MIDI computer Architecture. Only the solution with placement of the GEIRS and NRTS software on one computer is shown. The alternative two-computer solution is mentioned in [1].

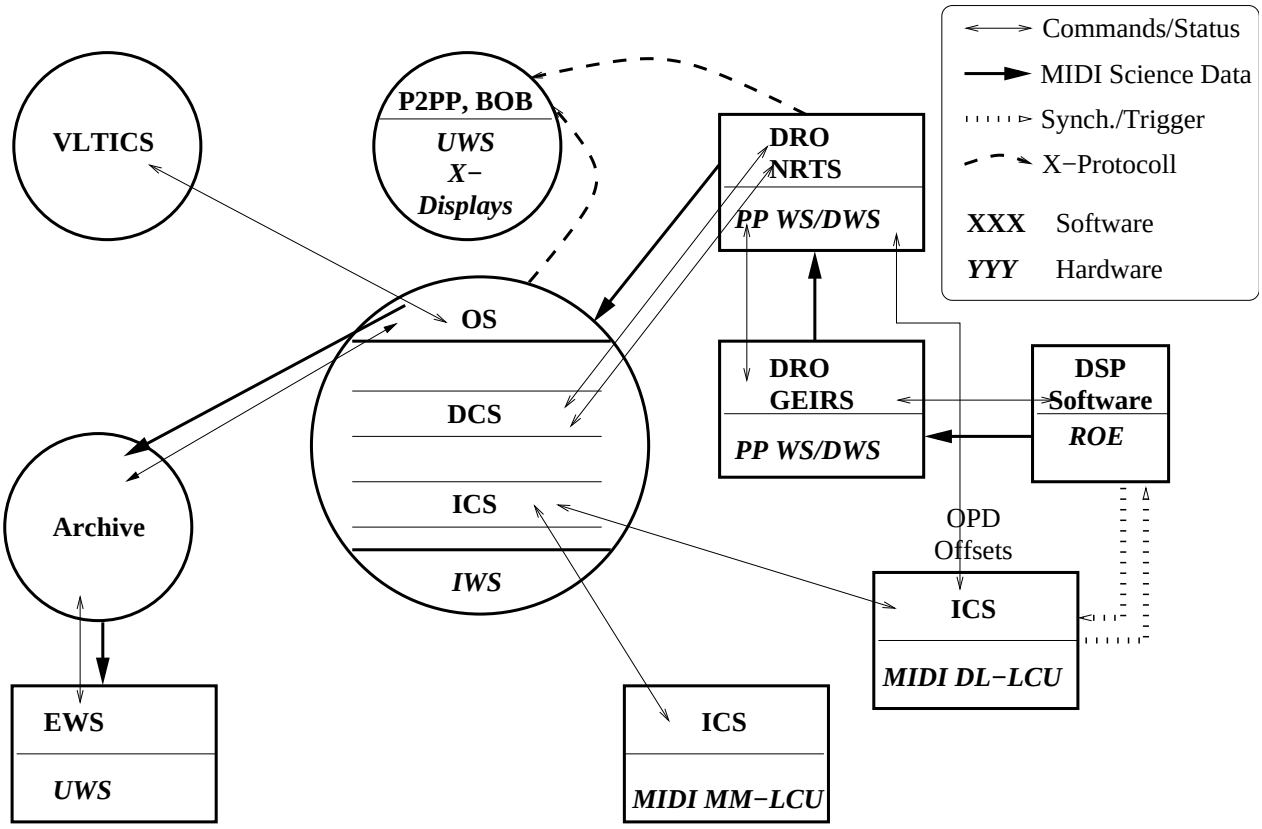

Figure 2: MIDI data and command/status flow.

3 USER MANUAL

3.1 Startup Tool

3.1.1 Interactive Startup

The recommended startup sequence in an interactive session:

1. ensure that the `DISPLAY` variable of the UNIX shell is defined

   ```
   wmidi > echo $DISPLAY
   ```

2. optionally start the OS control panel Fig. 4

   ```
   wmidi > miopanEngineering &
   ```

3. start ICS either with

   ```
   wmidi > miinsStart -proc ICS
   ```

   or with the `STARTUP` button of Fig. 4. This should result in ICS being `LOADED`.

4. start DCS either with

   ```
   wmidi > miinsStart -proc DCS
   ```

   or by pushing the `STARTUP` button of DCS in Fig. 4.

5. Start the VLTI interface with

   ```
   wmidi> miinsStart -proc VLTI
   ```

6. start OS either with

   ```
   wmidi > miinsStart -proc OS
   ```

   or by pushing the `STARTUP` button of OS in Fig. 4.

7. bring all subsystems into the `ONLINE/IDLE` states with either

   ```
   wmidi > msgSend "" mioControl online ""
   ```

   or by pushing the `ONLINE` button of OS in Fig. 4.

The less experienced user should simply set the `DISPLAY` and use

```
wmidi> miinsStart
```

which starts the instrument software plus a default set of GUI's with slightly less selective control.

3.1.2 Interactive Shutdown

The shutdown process follows the opposite ordering:

1. `wmidi > miinsStop -proc OS`
2. `wmidi > miinsStop -proc VLTI`
3. `wmidi > miinsStop -proc DCS`
4. `wmidi > miinsStop -proc ICS`
5. Close the remaining GUI's with their "Quit" selector.

The quick procedure is

`wmidi> miinsStop`

with no command line arguments, which also takes care of closing the light path by moving the FILT and SHUT to their CLOSED positions.

3.1.3 Processes

Starting and stopping of instrument subsystems (OS, ICS, DCS, VLTI) is supported through software based on the Startup Tool [8]. The UNIX command line calls on the Instrument Workstation are

`wmidi > miinsStart -proc OS`

to start the OS server processes `mioControl` and `bossArchiver_mio`,

`wmidi > miinsStart -proc ICS`

to start the ICS server processes `miiControl` and `miSimControl`,

`wmidi > miinsStart -proc DCS`

to start the DCS server processes on `wmidi` and `wminrts`, which comprise the "split server" `midControl`, the NRTS server `minrtsControl`, and the DRO server `midroControl`.

`wmidi > miinsStart -proc VLTI`

to start the VLTI simulation servers [9].

3.1.4 Graphical Interfaces

Graphical User Interfaces are created by either following push-buttons on other Graphical User Interfaces or calling them by name. Some of them are also bound to the `ctoo` STARTUP configuration; whence

- the standard ESO BOB panel [10] can be called by either

  `wmidi > bob &`

  or by

  `wmidi > miinsStart -panel BOB &`

- the standard ESO LOG Monitor panel can be called by either

  ```
  wmidi > logMonitor &
  ```

  or by

  ```
  wmidi > miinsStart -panel LOG &
  ```

- the standard ESO `vltisim` Control panel [9] can be called by either

  ```
  wmidi > vltisimcon &
  ```

  or by

  ```
  wmidi > miinsStart -panel VLTI &
  ```

- Fig. 3 can also be called by

  ```
  wmidi > miinsStart -panel OS_CONTROL &
  ```

- Fig. 4 can also be called by

  ```
  wmidi > miinsStart -panel OS_ENGINEERING &
  ```

- Fig. 5 can also be called by

  ```
  wmidi > miinsStart -panel ICS &
  ```

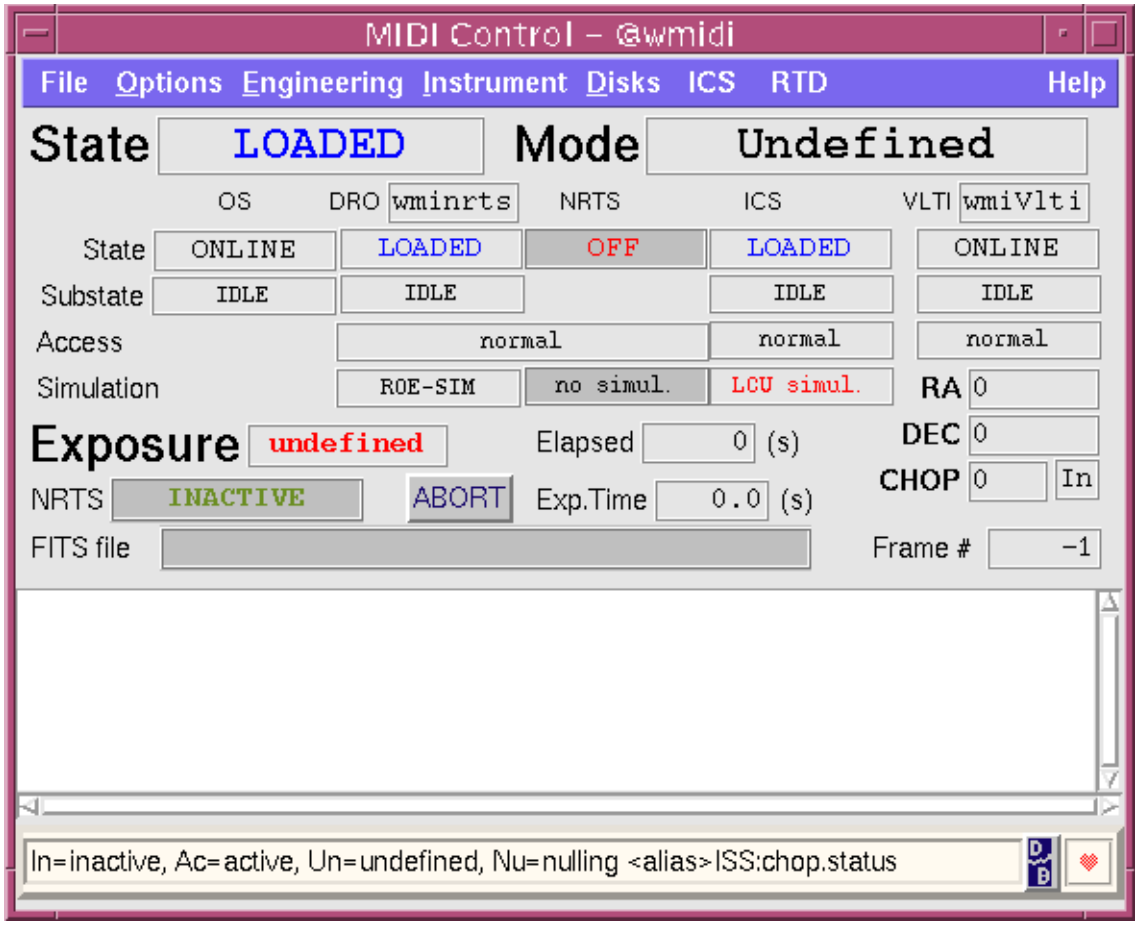

Figure 3: miopanControl - a slightly modified ESO standard. Buttons are described in the man page `miopanControl(1)`. Reproduced from [2].

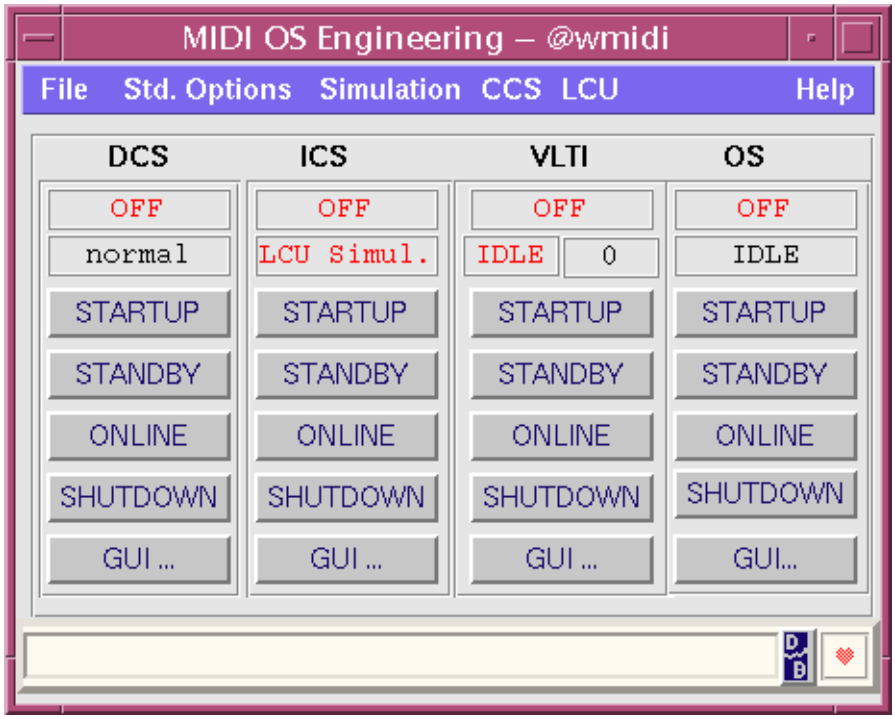

Figure 4: miopanEngineering - a slightly modified ESO standard panel accessible also with the `Engineering` option on the top bar of Fig. 3. See the man page `miopanEngineering(1)`. Reproduced from [2].

Figure 5: miipanControl - pops up if the `GUI...` button in the `ICS` column of the GUI in Fig. 4 is pressed. The further tree of GUI's hiding behind the `mimpan` and `mimalign` buttons is shown elsewhere [3]. See `miipanControl(1)`.

3.2 FAQ and Troubleshooting

3.2.1 msql Connectivity

On Paranal, the ACC database on the workstation `$ACC_HOST` tends to become unstable, which emits error messages like

```
environment_loader (pecs 000): ERROR: bootstrapping failed
environment_loader (pecs 000): ERROR: bootstrapping failed
```

if one logs in to `wmidi`. Other symptoms are that

```
wmidi > vccFastShow
```

does not show the complete lists of some hundred workstations and LCU's, or that the shutdown procedures of Sect. 3.1.2 fail. From the MIDI point of view there is nothing one can do but to call the support hotline to have the msql daemon restarted. (It is clear from a UNIX administrator's point of view that an extra user ought be created on the `$ACC_HOST` which—instead of getting an ordinary shell upon login—just initiates the revival of the daemon.)

If `miinsStop` does not work due to this ACC problem, the shutdown of Sect. 3.1.2 must be done in more irregular ways:

- shutdown ICS with

  ```
  wmidi> icbStop MIDI
  ```

- shutdown VLTI (only if the simulation `wmiVlti` is active)

  ```
  wmidi> vltisimStop
  wmidi> ccsShutdown -e wmiVlti
  ```

- shutdown OS with

  ```
  wmidi> msgSend "" mioControl exit ""
  wmidi> msgSend "" bossArchiver_mio exit ""
  ```

- shutdown DCS with

  ```
  wmidi> msgSend "" midroControl off ""
  wmidi> msgSend "" midroControl exit ""
  wmidi> msgSend "" midControl exit ""
  wminrts> minrtsStop
  wminrts> middiStop
  wminrts> vccEnvStop -e wminrts
  ```

4 MAINTENANCE MANUAL

4.1 Installation

4.1.1 Instrument Workstation

The standard installation of the OS, ICS and the DRO part of the DCS software on the IWS uses one of various options of `pkginBuild` [11] to retrieve and compile the source codes as layed out in the files `miins/config/miinsINSTALL*.cfg` as `midimgr`.

1. ```
   wmidi> cd ~/MIDI_SOURCES/${UNAMEN}
   wmidi> mv $INS_ROOT $INS_ROOT.‘date ’+%Y-%m-%dT%X’‘
   wmidi> mv $INTROOT $INTROOT.‘date ’+%Y-%m-%dT%X’‘
   wmidi> cmmCopy vltisimBUILD
   wmidi> pkginBuild vltisimBUILD -fromstep RETRIEVE -tostep INSTALL_FILES
   wmidi> export PKGIN_BUILD=~/MIDI_SOURCES/${UNAMEN}/miins ;# define source directory if not yet
   wmidi> cd $PKGIN_BUILD/..
   wmidi> cmmCopy miins
   wmidi> pkginBuild -step RETRIEVE
   ```

2. If the modules `slx`, `oslx` or `cfitsio` are in the list of modules in `miinsINSTALL_VLTSW.cfg`, it may be advisable to compile these first individually, because `pkginBuild` uses shared libraries created by these modules, and these cannot be replaced at `pkginBuild` runtime (under HP-UX).

3. ```
   wmidi> pkginBuild -fromstep BEFORE_INSTALL -tostep BUILD_MOD
   ```

   Ensure that `miinsEnvSet(1)` is called, which is done via the PECS scripts by logging in again...

   ```
   wmidi> logout
   wmidi> rlogin wmidi -l midimgr
   wmidi> pkginBuild -fromstep BUILD_ENV
   ```

4. After a re-installation of the user `midi` on the IWS, one must recreate `wmidi:~midi/.netrc` to allow `ftp(1)` file transfer to the DWS with `mioSetupDistr(1)`, optionally including information for `mioFtp2Ews(1)`. In this case it is also strongly recommended to establish symbolic links in the `~midi/.pecs` directory as already established at that time in `~midimgr/.pecs`. For the equivalent reason (update the FITS template files on the IWS), one must recreate `wminrts:~midimgr/.netrc` to allow `ftp(1)` file transfer in the other direction. (At the time of this writing, there is a slight asymmetry here, because the transfer is done from DWS at compile time by `midimgr`, but from IWS at startup time by `midi`.)

5. If p2pp is to be used on Paranal, the instrument package must also be uploaded to the `wvgdhs` workstation with

   ```
   wmidi> cd $PKGIN_BUILD/../miotsf/src
   wmidi> make install_ip
   ```

   and following the instructions, including the temporary run of p2pp. This installs the instrument package on `wvgdhs:~instmgr/instruments`, in conjunction to a reference to this directory in `wvgdhs:~visitor/.p2pp.cf`.

   On the `wmidi` test machine at the MPIA it is useful to install the user `instmgr` on `wmidi` according to [12] to simplify this step (if not yet done): as `root`

```
wmidi# useradd -g vlt -d /diska/home/instmgr -s /bin/bash -c "P2PP Installation" -m instmg
wmidi# passwd instmgr
```

and after `rlogin` as `instmgr`

```
wmidi> /etc/pecs/bin/pecssh mklinks -i
wmidi> echo "wmidi midimgr" >> ~/.rhosts ; # edit for login as midi
wmidi> chmod go-r ~/.rhosts
wmidi> cd ~/.pecs
wmidi> ln -s /introot/config/miins-apps-wmidi.env apps-wmidi.env
wmidi> ln -s /introot/config/miins-misc-wmidi.env misc-wmidi.env
wmidi> ln -s /introot/config/miins-misc-wmidi.ali misc-wmidi.ali
```

4.1.2 Preprocessor Workstation

The equivalent standard installation of the NRTS software on the PP WS ensures that configuration files for schedules (see the man page `mioSched.ins(5)`) and for FITS templates (see the man page `mioSetup.fits`) are consistent. [Absence of cross-compilation facilities in the installation tool [11] means that this compilation of the PP WS *must* be done separately.]

```
wminrts> export PKGIN_BUILD=~/MIDI_SOURCES/${UNAMEN}/minrts ;# chose source directory if not y
wminrts> cd $PKGIN_BUILD/..
wminrts> mv $INS_ROOT $INS_ROOT.‘date ’+%Y-%m-%dT%X’‘;# not needed with pkginBuild 1.75 and INS
wminrts> mv $INTROOT $INTROOT.‘date ’+%Y-%m-%dT%X’‘;# not needed with pkginBuild 1.75 and INSTA
wminrts> cmmCopy minrts
wminrts> pkginBuild -fromstep RETRIEVE
```

Note that some of the DCS installation, the GEIRS part, is not covered by `pkginBuild` as described above, and needs additional action as described in [1].

After a re-installation of the user `midi` on the DWS, one must recreate `~midi/.rhosts` to allow `remsh(1)` access from the IWS account `midi`.

4.1.3 Finalize

Since this installation is done after login as `midimgr` on these computers, and since `pkginBuild` finally starts the environments on these computers under the same account, it is advisable to stop the workstation environments afterwards.

```
wmidi > vccEnvStop -e wmidi
wmidi > vccEnvStop -e wmiVlti
wminrts > vccEnvStop -e wminrts
```

Otherwise a subsequent attempt to run the environments as the ordinary `midi` user may result in problems with file permissions.

4.1.4 Secure Shell

For the test environment behind a firewall, it may also be necessary to install the secure shell to allow remote access to the machine. This does *not* make sense on Paranal, but is usually done at MPIA. As root:

1. Copy `openssl-0.9.6i.tar.gz` from `http://www.openssl.org/source` and `openssh-3.6.1p2.tar.gz` from `http://www.openssh.com/portable.html` to `/vltdata/tmp`. Unpack

   ```
   cd /vltdata/tmp
   gunzip openssl-* ; tar xf openssl-*
   gunzip openssh*.gz ; tar xf openssh*.tar
   ```

2. ```
   cd /vltdata/tmp/openssl-0.9.6i
   PATH=/vlt/*20??/gnu/bin:$PATH
   ./config
   make
   make install
   ```

3. Only on the IWS (for HP-UX): Ensure the zlib is found:

   ```
   wmidi> export LDFLAGS="-L/vlt/APR2003/tcltk/lib"
   ```

   Note that the exact library path depends on the current VLTCS version.

4. ```
   cd /vltdata/tmp/openssh-*p?
   CFLAGS="-I/vlt/APR2003/tcltk/include" ./configure --with-utmpx \
       --without-pam --with-ssl-dir=/usr/local/ssl
   ```

   Note that the exact include-path depends on the current VLTCS version.

5. on the IWS (HP-UX)

   ```
   wmidi> vi config.h ; # and insert #define HAVE_UTIMES
   wmidi> useradd -d /home/sshd -s /bin/false -c "openssh daemon" sshd
   wmidi> PATH=$PATH:/vlt/*20??/gnu/hppa*/bin
   ```

   or alternatively on the DWS (Solaris)

   ```
   wminrts> useradd -d /dev/null -g bin -s /bin/false -c "openssh daemon" sshd
   wminrts> PATH=$PATH:/vlt/*20??/gnu/sparc*/bin
   ```

   (The `PATH` extension ensures that `strip` is found.) Then

   ```
   make
   make install
   ```

6. on the IWS (HP-UX)

   ```
   wmidi> cp ~midimgr/MIDI_SOURCES/${UNAMEN}/miins/src/sshd.HP /sbin/init.d/sshd
   wmidi> cd /sbin/rc2.d
   wmidi> ln -s ../init.d/sshd S180sshd
   wmidi> cd /sbin/rc1.d
   wmidi> ln -s ../init.d/sshd K170sshd
   ```

   or aternatively on the DWS (Solaris)

   ```
   wminrts> rcp wmidi:~midimgr/MIDI_SOURCES/${UNAMEN}/miins/src/sshd.SUN /etc/init.d/sshd
   wminrts> cd /etc/rc3.d
   wminrts> ln -s ../init.d/sshd S18sshd
   wminrts> cd /etc/rc2.d
   wminrts> ln -s ../init.d/sshd K25sshd
   ```

7. Edit `/usr/local/etc/sshd_config` to allow X11 forwarding.

   ```
   X11Forwarding yes
   X11DisplayOffset 10
   KeepAlive yes
   ```

8. Allow the execution of `ssh` on the DWS without prompting for the password.

Proper installation of the PECS templates in the earlier sections ensures that the `PATH` of the `midi` account on the IWS already contains the new executables in `/usr/local/bin`, but the DWS is not administered in an equivalent way which may call for additional configuration there.

4.1.5 P2PP

For the test environment at MPIA on the IWS, one ought also install P2PP [13]. This does *not* make sense on Paranal, because `p2pp` is not started on the IWS there any longer. As `vltmgr`:

1. Copy the "Unix distribution kit" from `http://www.eso.org/observing/p2pp/P2PP-tool.html` or `ftp://ftp.eso.org/pub/users/uss/ohs/jp2pp` to `/vlt/`. Note that it is *not* necessary to install the Java Runtime Environment, since this has already been installed in the process of installing the VLTCS [12]. Unpack

   ```
   wmidi> cd /vlt
   wmidi> gunzip p2pp*.gz ; tar xf p2pp*.tar
   ```

2. Follow the instructions in the `INSTALL*.html` file:

   ```
   wmidi> cd /vlt/p2pp-2.8
   wmidi> make
   wmidi> rm /vlt/p2pp-*.tar
   wmidi> mkdir -p /vlt/p2pp-2.8/cache/instruments
   wmidi> chmod g+wx /vlt/p2pp-2.8/cache/instruments
   wmidi> chmod g+wx /vlt/p2pp-2.8/cache
   ```

3. If the `midi` account has been re-created, also add the file `~midi/.p2pp.cf` with the lines

   ```
   CACHE.FOLDER "/diska/home/midi/p2pp/cache"
   IMPEX.FOLDER "/diska/home/midi/p2pp/impex"
   INSTRUMENTS.FOLDER "/diska/home/midi/p2pp/instruments"
   APPSERVER.LOG      "/diska/home/midi/p2pp/logs/appserver.log"
   ```

   and execute

   ```
   wmidi> mkdir -p ~/p2pp/cache ~/p2pp/instruments/MIDI
   ```

   as user `midi`.

4. Edit `/vlt/p2pp-2.8/config/site.cf` to penetrate the MPIA firewall:

   ```
   APPSERVER.HTTPPROXYHOST "web-proxy.mpia-hd.mpg.de"
   APPSERVER.HTTPPROXYPORT "3128"
   ```

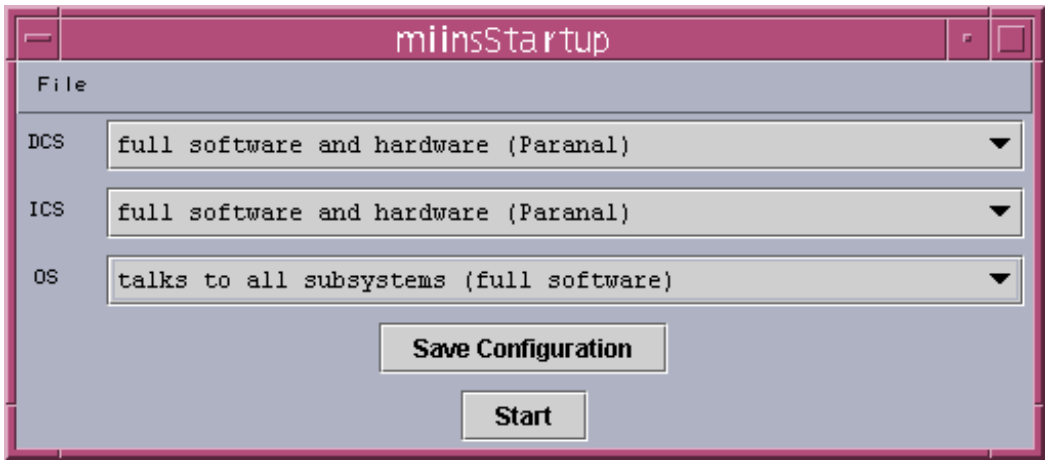

Figure 6: The graphical user interface accessible with the `miinsStartup(1)` command.

and optionally

```
DEFAULT.PRINTER "laps_u"
```

5. Ensure that each start of p2pp works with the latest instrument package. This is achieved by adding the following ten lines before the `# Standard settings` in `/vlt/p2pp-*[0-9]/bin/p2pp`:

```
# at MPIA ensure that the midi account uses the most recently installed TSF's
case $USER in
        midi | instmgr)
                ifold=`fgrep INSTRUMENTS.FOLDER ~/.p2pp.cf | awk -F \" '{print $
2}'`
        (cd $ifold/MIDI ; rm *.tsf; ln -s $INS_ROOT/SYSTEM/COMMON/TEMPLATES/TSF/
MIDI*.tsf .)
        ;;
esac
```

and adding two links,

```
wmidi> ln -s $INTROOT/config/MIDI.isf $INTROOT/config/MIDI.zip ~/p2pp
```

Proper installation of the PECS templates in the earlier sections ensures that the `PATH` of the `midi` account on the MPIA IWS already contains the new executables in `/vlt/p2pp*/bin`.

4.2 Startup Reconfiguration

The UNIX command line call on the Instrument Workstation to edit some global configuration parameters in the `mimcfgINS.cfg` and `mimcfgSTART.cfg` configuration files is

```
wmidi > miinsStartup
```

which opens a GUI shown in Fig. 6.

4.3 FAQ and Troubleshooting

4.3.1 OS keeps talking to wmiVlti instead of wvgvlti

Symptom After modifying the `OCS.TEL.ENVNAME` in `mimcfgINS.cfg`, recompilation with `make clean all man install` in `mimcfg/src`, and re-starting the OS servers, OS still seems to forward commands to the VLTI server in the "old" environment.

Main Cause Behaviour of some parts of OS and the BOB setup is not only steered by the configuration keywords that are read once when OS is started, but also through additional parameters of the UNIX shell. It also seems that `mioControl`, as it is started as a subprocess of RTAP, inherits shell environment variables from RTAP, and needs to be started under a `wmidi` fullCCS environment that already keeps the correct shell variables.

Workaround Follow a proper sequence of actions if the VLTI environment is to be changed:

- Shutdown the old MIDI processes on `wmidi` with `miinsStop(1)`, including the VLTI.
- Stop the CCS environment with

  ```
  wmidi > vccEnvStop -e wmidi
  ```

  and close all BOB windows.
- Edit `OCS.TEL.ENVNAME` in `mimcfg/config/mimcfgINS.cfg` and distribute with `cd ../src ; make clean all man install`
- Logout and login again on all terminals on the IWS to have `osbEnvSet(1)` redefine `TCS_ENVNAME` and `VLTI_ENVNAME`.
- Re-start the CCS environment and MIDI SW with

  ```
  wmidi > miinsStart
  ```

The script `miinsSwitchVlti` in `miins/test` is tailored to support this cycle. A more advanced proposal was made in VLTSW20030001, but was rejected by ESO.

Second Cause Someone tried to be smart and created subordinate copies of the `mimcfgINS.cfg` file to avoid editing and moved them around, which dodged the `Makefile` idea of which files are older than others.

Workaround Use only one configuration file `mimcfgINS.cfg`, edit it only in the source code directory and always run `make` there.

Third Cause It is actually not the OS server but one of the sequencer (template) scripts that uses a hardcoded environment name. This error usually means that a programmer tried to bypass the `isstcl` interface library in these scripts, but this would anyway be out of the scope of this document.

4.3.2 Scan Link Setup at VLTI Environment Switching

Symptom In some cases, especially when switching between `wmiVlti` and `wvgvlti` environments, the `miinsStart` reports problems with setting up the scan links between `wmidi` and `$VLTI_ENVNAME`.

Cause

Workaround In these cases one should explicitly close the connection with

```
wmidi> miinsStop -e VLTI
```

then call

```
wmidi> scanei
```

to clean/erase the scan tables towards `wmiVlti` and `wvgvlti`, then try again

```
wmidi> miinsStart -e VLTI
```

If the error is close to "Failed to add or delete an entry CE function for attribute ... already existing (line...)," this will not help and one must re-build the databases of the environments (as `midimgr`):

```
wmidi> pkginBuild -fromstep BUILD_ENV -env wmidi
wmidi> pkginBuild -fromstep BUILD_ENV -env wmiVlti
```

4.3.3 "missing separator" make messages on wminrts

Symptom Compiling NRTS modules or VLTSW on wminrts creates errors of the form

```
/tmp/vltMake_3420_midimgr:28: *** missing separator.  Stop
```

Cause The environment variable `CPU` is probably not set. See VLTSW20000323 and VLTSW20030174.

Workaround Set the environment variable

```
export CPU=PPC604
```

either in the local PECS files `apps-$HOST.env` of all relevant parties (`midimgr, vltmgr,...`) or better in the global configuration file of the `~pecsmgr/releases/000/etc/locality/apps-$HOST.env`.

4.3.4 pkginBuild errors compiling `miopub`

Symptom Recompilation of the `miopub` module with `pkginBuild` returns an `ERROR`.

Cause As one can see in the `$MIDISW/INSTALL/pkginBuild.err` file, some shared libraries cannot be replaced with `rm -rf` and `cp`; the error message is `Text file busy` as described in `unlink(2)`. This probably means that `pkginBuild` is using `oslx` which is linked to `cfitsio`. In a ab initio compilation without any prior files in `$INT_ROOT/lib`, this problem does not show up.

Workaround Execute the `make clean all man install` from the UNIX shell in the module, not via `pkginBuild`, or remove `$INT_ROOT/lib/libcfitsio.sl` manually before starting `pkginBuild`.

4.3.5 NRTS environment cannot be started

Symptom After first installation of the instrument on Paranal in Nov. 2002, the NRTS environment `wminrts` could not be started on the workstation `wminrts` after rebuilding everything from scratch. Some of the standard processes listed in `CcsEnvTable` core dumped.

Cause The VLTCS, including the binaries, had just been copied from another computer of a different type than MIDI's ULTRA Enterprise 450 Server.

Workaround Install the VLTCS properly according to the manual [12], step by step.

(Note that, though [1] tries to distribute a different idea in some of its sections that deal with GEIRS, the MIDI DWS is *not* what SUN calls `SPARCstation` or `UltraSparc` computers. It is *only* the four processors (SPARC-II) which carry this name.)

4.3.6 No logFile on wminrts

Symptom The files `logAuto` and `logFile` in the `/vltdata/tmp` directory on the DWS do not exist and apparently have been removed.

Cause This seems to be a rare event caused by the periodic invocation of `logVLTBackup(1)` [see the `crontab(1)` of the account `vlt`] in conjunction with a high log level of the NRTS processes.

Recovery Remake (empty) `logFile` and `logAuto` as described in [12, §3.4.2]. As root, restart the logger as described in [12, §3.4.2]. (If this is impractical, reboot the NRTS computer.)

4.3.7 hisDHMngr errors at environment start

Symptom At start time of the `RTAPENV`, the `logFile` reports warnings of the kind

```
... wmidi his hisDHMngr 214 62 hisControlPoint.c 1005 5 1 S ccsERR_HIS_CONTROL_TABLES :
        No setup can be enabled for history table with PLIN 47
... wte35 his hisDHMngr 214 62 hisDHMngr.c 1002 5 2 W ccsERR_HIS_CHECK_TABLES :
       Point PLIN '47' could NOT be set to IDLE/ARMED
```

for two different PLIN's.

Cause The corresponding setup functionality is not enabled in `CCS/ccs/dbl/CCSMON.db` for the two OLDB points `:ccs_config/monitoring/trend[12]`. (The association of PLIN's with OLDB points can be deduced from `$VLTDATA/ENVIRONMENTS/$RTAPENV/ccs_log`.)

Workaround Ignore. This seems not to be an error but a feature of the CCS monitor in combination with a diligant logging within the `his` module.

4.3.8 Why are the module sources not at the standard place?

Question Why are the MIDI modules not in the `MIDI_SOURCES` directory but one store lower at `MIDI_SOURCES/$UNAMEN`?

Answer The computer disk layout in the Garching VLT Control Model (and since Aug 2004 also on Paranal) cross-mounts the home-directory of the `midimgr`, which means that `~midimgr` refers to the *same* directory on the HP-UX workstation and the Solaris workstation (see also VLTSW20020194). If one would not separate the modules, the binaries of all modules that are used on `wmidi` as well as on `wminrts` (i.e., `mio`) would be overwritten by `pkginBuild` in an unpredictable way; this would create a mess of binaries at installation time and confuse the makefile dependency generator.

This precaution would not be needed at MPIA.

4.3.9 Parameter Ranges for Use with Templates

Question Where are the allowable parameters for use with templates?

Answer The recognized keyword value lists are at their standard place, the instrument summary file, `miotsf/config/*.isf`.

- The motor positions names (including filters) and instrument modes in this file should be kept up-to-date with the contents of `mimcfg/config/mimcfgINS.cfg`.
- The list of pre-defined schedule files and windowing files should be updated with the file lists in the `INS_ROOT` directory on the IWS (as documented in the ISF itself) after going online with the software. Most of the windowing files are generated dynamically during this process as documented in [14]. It is highly recommended that the template software does this also dynamically instead of being based on static lists.
- Parameters that cannot be changed by software (example: the detector gains) should not be used by the templates and therefore not be mentioned.
- The telescope and VLTI DL keywords need to be updated with whatever has been documented in the ISS modules—this is certainly *not* part of the MIDI software and therefore not to be documented here. It obviously indicates bad design of P2PP that this information must be kept in an instrument specific file (VLTSW20040302).

Additional documentation of some of these is in the comments of the dictionary entries (module `dicMIDI`). This includes documentation of keywords with specific values that trigger automatic choices by the subsystems themselves.

4.3.10 CMM modules

Question Which are the MIDI CMM modules?

Answer The software modules are obviously listed at the standard places used by `pkginBuild`, which is `$PKGIN_BUILD/config/miinsINSTALL.cfg` on `wmidi` and `$MIDI_SW/minrts/config/minrtsINSTALL.cfg` on `wminrts`. The software documentation is generally kept in the `doc` subdirectory of these modules, but additional copies exist in the CMM modules VLT-TRE-MID-15824-0262, VLT-TRE-MID-15824-0264, VLT-TRE-MID-15823-0266, VLT-TRE-MID-15823-0268, and VLT-TRE-MID-15824-0270.

All these are subject to change; the template type of modules `mi?seq` and `mi?tsf` have been triplicated recently, for instance, rendering it almost impossible to figure out the currently active set.

5 ISSUES UNRELATED TO MIDI SOFTWARE

5.1 FAQ and Troubleshooting

5.1.1 man(1) does not work under Solaris

Symptom The `man(1)` command does not work under SunOS.

Cause This seems to occur when the machine is rebooted, as the actions in `/etc/rc2.d/*RMTMPFILES` may end up in missing write permissions for the `/tmp` directory.

Recovery Restore write permissions for everyone to `/tmp`.

5.1.2 CMM fails under Solaris

Symptom The CMM commands [15] do not work after re-installation of the operating system and VLTCS on the Solaris computer.

Cause Probably a missing `/etc/defaultrouter` or `/etc/resolv.conf` or an error in `/etc/inet/netmasks`.

6 APPENDIX

References

[1] Udo Neumann and Clemens Storz. *MIDI Detector Control Software, User and Maintenance Manual*, 20 October 2005. VLT-TRE-MID-15823-0268.

[2] Richard J. Mathar. *MIDI Observation Software, User and Maintenance Manual*, 24 November 2004. VLT-TRE-MID-15824-0264.

[3] Udo Neumann. *MIDI Instrument Control Software, User and Maintenance Manual*, 16 June 2004. VLT-TRE-MID-15823-0266.

[4] S. Hippler. *MIDI Computer System Architecture*, 30 January 2001. VLT-TRE-MID-15823-0127.

[5] Richard J. Mathar. *MIDI Instrument Control Software*, 1 February 2001. VLT-TRE-MID-15823-0128.

[6] C. Storz. *MIDI Detector control software*, 7 February 2001. VLT-TRE-MID-15823-0129.

[7] W. Jaffe. *Overview of MIDI Software Systems*, 31 January 2001. VLT-TRE-MID-15824-0130.

[8] D. Popovic. *VLT Software, INS Common Software, Startup Tool User Manual*, 26 August 2005. VLT-MAN-ESO-17240-2153.

[9] N. Housen. *Simulation of VLTICS for Instrumentation SW, User Manual*, 1 December 2001. VLT-MAN-ESO-15410-2372.

[10] E. Allaert. *VLT Software - HOS/Broker for Observation Blocks User Manual*, 24 April 2007. VLT-MAN-ESO-17220-1332.

[11] S. Skole and R. Schmutzer. *VLT Software, Installation Tool for VLT SW Packages, User and Maintenance Manual*, 31 December 2005. VLT-MAN-ESO-17240-1913.

[12] S. Feyrin. *VLT Common Software - Installation Manual*, 21 February 2007. VLT-MAN-ESO-17200-0642.

[13] F. Comerón and D. Silva. *Phase 2 Proposal Preparation Tool (P2PP), User Manual*, 6 edition, 20 December 2004. VLT-MAN-ESO-19200-1644.

[14] J. de Jong. *MIDI Near-Real Time Software, User and Maintenance Manual*, 8 February 2004. VLT-TRE-MID-15824-0270.

[15] P. Sivera. *VLT Software - Configuration Management Module User Manual*, 22 October 2001. VLT-MAN-ESO-17200-0780.